



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/625,812	07/26/2000	Timothy J. Van Hook	00100.00.0105	8263
29153 7590 12/11/2007 ADVANCED MICRO DEVICES, INC. C/O VEDDER PRICE KAUFMAN & KAMMHOLZ, P.C. 222 N.LASALLE STREET CHICAGO, IL 60601			EXAMINER HSU, JONI	
			ART UNIT 2628	PAPER NUMBER
			MAIL DATE 12/11/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

09/625,812

Applicant(s)

VAN HOOK, TIMOTHY J.

Examiner

Joni Hsu

Art Unit

2628

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 28 September 2007.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-18,23,25-31 and 33 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 25-31 is/are allowed.
- 6) ☒ Claim(s) 1-18,23 and 33 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on September 28, 2007 has been entered.

Response to Arguments

2. Applicant's arguments with respect to claims 1-13 and 23 have been considered but are moot in view of the new ground(s) of rejection.

3. Applicant's arguments, see p. 9, filed September 28, 2007, for rejections of claims 1-13 under 35 U.S.C. 103(a) are considered and are persuasive. So, rejection has been withdrawn. But, upon consideration, new ground of rejection is made in view of Joy (US006341347B1).

4. Applicant argues Joffe (US006330584B1) does not teach execution pipeline having depth less than or equal to plurality of programs, each program includes plurality of instructions (p. 9).

In reply, Examiner agrees. However, new grounds of rejection are made in view of Joy.

5. Applicant's arguments filed September 28, 2007, with respect to Claims 14-18 and 33 have been fully considered but they are not persuasive.

6. As to Claims 14 and 23, Applicant argues Krishna (US006161173A1) teaches no-op being inserted in pipeline, and so does not teach execution instructions amongst plurality of end programs wherein no no-op is inserted into pipeline for purposes of insuring instruction is completed before execution of another instruction from another program (p. 10).

Art Unit: 2628

In reply, Examiner points out Applicant's disclosure describes inserting no-ops into instruction stream or retarding launching of new programs until first program finishes (p. 17, ll. 8-11). So, when no no-op is inserted, that means 1st instruction is completed and execution of 2nd instruction can begin. Krishna teaches local scheduling circuitry stops main scheduler from issuing selected operation if latency of another operation would create conflict with main scheduler issuing selected operation (c. 2, ll. 56-60). So, it is ensured that 1st instruction is completed before beginning execution of 2nd instruction. Information in each entry describes either no-op or associated operation which is to be executed (c. 5, ll. 36-38). So, when no-op is inserted, this means operation is not ready to be executed. When associated operation which is to be executed is inserted, there is no no-op inserted, this means associated operation is ready to be executed, meaning 1st instruction is completed and it is now okay to execute associated operation. So, Krishna teaches no no-op is inserted for purpose of ensuring that 1st instruction is completed before beginning execution of 2nd instruction.

7. As to Claim 33, Applicant argues Joffe teaches resource is not provided to task until after all tasks sharing resource has finished accessing resource. Task may be uncompleted but resource is allocated. So, Joffe does not teach checking if all programs are completed (p. 10).

In reply, Examiner states Joffe teaches if task attempts to access unavailable resource, task is suspended. When resource becomes available, suspended task is resumed, and instruction accessing resource is re-executed. Task does not get access to same resource until after every other task sharing resource has finished accessing resource (c. 2, ll. 29-39). If Wait signal is asserted, instruction execution is not completed and PC register is frozen, but task remains active, and instruction will be executed again starting next clock cycle. If Suspend/Wait signals

are deasserted, the PC register is changed to point to next instruction (c. 10, ll. 19-32). So, Joffe teaches checking to see if Suspend/Wait signals are desasserted, which indicates instruction execution is completed (c. 10, ll. 19-32) and resource has become available (c. 2, ll. 32-34), and resource becomes available after every other task sharing resource has finished accessing resource (c. 2, ll. 29-39). So, Joffe teaches checking to see if all of the programs are completed.

Claim Objections

8. Claims 15-18 objected to because of the following informalities: Claims 15-18 each recite "The method of claim 14 or claim 24 further including...". Claim 24 has been cancelled, and therefore Claims 15-18 cannot depend from Claim 24. Appropriate correction is required.

Claim Rejections - 35 USC § 112

9. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

10. Claim 17 is rejected under 35 U.S.C. 112, 2nd paragraph, as being indefinite for failing to particularly point out and distinctly claim subject matter which applicant regards as invention.

11. Claim 17 recites the limitation "said **graphics** processing execution pipeline". There is insufficient antecedent basis for this limitation in the claim.

Claim Rejections - 35 USC § 103

12. Text of sections of Title 35, U.S. Code 103(a) not included can be found in prior action.

13. Claims 1-7, 9-11, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Joy (US006341347B1) in view of Krishna (US006161173A).

14. As per Claim 1, Joy teaches programmable processor for executing plurality of programs (threads), each of plurality of programs has plurality of instructions (c. 8, ll. 33-39). Multiple-

Art Unit: 2628

thread execution pipeline includes pipeline stages. Each pipeline stage includes flip-flops, and each flip-flop is coupled to select-bus lines selecting active thread from among plurality of execution threads. This allows each pipeline stage to immediately switch from first thread to second thread when first thread stalls, so second thread is executing on otherwise unused or idle pipeline stage. This allows for execution of more threads without increasing number of pipeline stages (c. 7, ll. 41-44; c. 8, ll. 14-26; c. 10, ll. 14-37; c. 37, ll. 9-23). Since each pipeline stage selects one active thread, this means there are a same number of pipeline stages as there are threads (programs). So, programmable processor has execution pipeline having depth less than or equal to plurality of programs. Since each pipeline stage switches from executing instructions from first thread to executing instructions from second thread when first thread stalls, and later resuming execution of instructions of postponed stalling first thread (c. 3, ll. 14-25; c. 7, ll. 41-45; c. 8, ll. 27-39), this means there is interleaver for interleaving instructions from plurality of programs and providing instructions to pipeline for execution such that number of plurality of programs that are interleaved is greater than or equal to depth of pipeline.

However, Joy does not explicitly teach execution pipeline has average pipeline latency of one instruction per cycle. However, Krishna teaches scheduler allocates fixed latency, which is typically one clock cycle, between issuing instruction to execution pipeline and execution pipeline returning result (c. 4, ll. 1-4). For some instructions, execution pipeline has longer latency (c. 4, ll. 4-5). Since fixed latency is typically one clock cycle for one instruction, Krishna is considered to teach execution pipeline has average latency of one instruction per cycle.

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify device of Joy so pipeline has average latency of one instruction per cycle as

Art Unit: 2628

suggested by Krishna because it results in a more streamlined pipeline operation and simplified design (Krishna, c. 2, ll. 60-67). Even though Joy does not teach pipeline (Joy, c. 37, ll. 9) has average latency of one instruction per cycle, Krishna teaches it is typical for instructions in pipeline to have average latency of 1 instruction per cycle. So, it would be obvious that pipeline of Joy can be used to execute instructions that have average latency of 1 instruction per cycle.

15. As per Claim 2, Joy teaches that the pipeline has a datapath with a depth equal to the number of programs (c. 10, ll. 14-37; c. 37, ll. 9-23).

16. As per Claim 3, Joy does not teach next instruction from one program is not provided to pipeline until previous instruction of one of the programs has completed. But, Krishna teaches local scheduling circuitry stops main scheduler from issuing selected operation if latency of another operation would create conflict with main scheduler issuing selected operation (c. 2, ll. 56-60). So, next instruction is not provided to pipeline until previous instruction has completed.

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify device of Joy so next instruction from one of plurality of programs is not provided to pipeline until previous instruction of one of plurality of programs has completed because Krishna suggests sometimes latency of another instruction would create conflict with main scheduler issuing selected instruction, so in order to avoid this conflict, next instruction is not provided to pipeline until previous instruction has completed (c. 2, ll. 56-60).

17. As per Claim 4, Joy teaches each program of the plurality of programs is independent of the other of the plurality of programs (c. 3, ll. 14-25).

18. As per Claim 5, Joy teaches interleaving instructions (c. 3, ll. 14-25; c. 7, ll. 41-45; c. 8, ll. 27-39), and so instructions are executed out of order.

Art Unit: 2628

However, Joy does not teach output buffer for storing out of order data output. However, Krishna teaches execution engine (140, Fig. 1) has an out-of-order architecture (c. 5, ll. 11-12), and scheduler (150) receives results from execution units (170, 175, 180) and stores results (c. 5, ll. 28-35). So, Krishna inherently teaches output buffer for storing out of order data output.

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify device of Joy to include output buffer for storing out of order data output because Krishna suggests since instructions are executed out of order (c. 5, ll. 11-12), output buffer is needed to store out of order data output so data can put in correct order (c. 5, ll. 28-35).

19. As per Claim 6, Joy discloses one or more of a register copy, program counter, and program counter stack provided for each of the plurality of programs (c. 6, ll. 34-36).

20. As per Claim 7, Joy teaches one of control/computing resources, instructions, instruction memory, data paths, data memory, caches are shared by plurality of programs (c. 8, ll. 59-66).

21. As per Claim 9, Joy teaches instructions for loading data from memory (c. 8, ll. 59-67).

22. As per Claim 10, Joy teaches instructions for storing data in memory (c. 8, ll. 59-67).

23. As per Claim 11, Joy discloses that the data memory comprises a cache (c. 8, ll. 59-67).

24. As per Claim 23, Joy teaches programmable processor for executing plurality of programs, programmable processor having execution pipeline having depth less than or equal to plurality of programs wherein each of plurality of programs has plurality of instructions; and interleaver for interleaving instructions from plurality of programs and providing instructions to pipeline for execution, as discussed in the rejection for Claim 1.

However, Joy does not teach execution pipeline has average pipeline latency of one instruction per cycle; and next instruction from one of plurality of programs is not provided to

Art Unit: 2628

pipeline until a previous instruction of one of plurality of programs has completed and wherein no no-op is inserted into pipeline for purpose of ensuring next instruction is not provided to pipeline until previous instruction has completed. But, Krishna teaches scheduler allocates fixed latency, which is typically one clock cycle, between issuing instruction to pipeline and pipeline returning result (c. 4, ll. 1-4). For some instructions, pipeline has longer latency (c. 4, ll. 4-5). Since fixed latency is typically one clock cycle for one instruction, Krishna is considered to teach execution pipeline has average latency of one instruction per cycle. This would be obvious for reasons for Claim 1. Applicant's disclosure describes inserting no-ops into instruction stream or retarding launching of new programs until 1st program finishes (p. 17, ll. 8-11). So, when no no-op is inserted, that means 1st instruction is completed and execution of 2nd instruction can begin. Krishna teaches local scheduling circuitry stops main scheduler from issuing selected operation if latency of another operation would create conflict with main scheduler issuing selected operation (c. 2, ll. 56-60). So, it is ensured 1st instruction is completed before beginning execution of 2nd instruction. So, next instruction is not provided to the pipeline until a previous instruction has completed. This would be obvious for reasons for Claim 3. Operation is executed if no no-op is inserted into pipeline (c. 5, ll. 36-38; c. 2, ll. 41-45). So, when no no-op is inserted into pipeline, this ensures first instruction is completed before beginning execution of second instruction. So, when no-op is inserted, this means that operation is not ready to be executed. When associated operation which is to be executed is inserted, there is no no-op inserted, this means that associated operation is ready to be executed, meaning 1st instruction is completed and it is now okay to execute associated operation. So, no no-op is inserted for purpose of ensuring 1st instruction is completed before beginning execution of 2nd instruction.

Art Unit: 2628

It would be obvious to one of ordinary skill in the art at the time of invention by applicant to modify Joy to include checking no no-op is inserted into pipeline for ensuring next instruction is not provided to pipeline until previous instruction has completed because Krishna suggests no-op is needed for indicating previous instruction has not yet completed (c. 5, ll. 26-38).

25. Claim 8 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joy (US006341347B1) and Krishna (US006161173A) in view of Nguyen (US005961628A).

Joy and Krishna are relied upon for the teachings for Claim 1. Joy and Krishna implicitly teach SIMD execution of vector instructions without addressing vector lengths.

But, Joy-Krishna do not teach executing SIMD vector instructions of length N and executing in parallel instructions having SIMD lengths that sum up to N. However, Nguyen teaches processor executes SIMD vector instructions of vector length N and executes in parallel plurality of instructions having SIMD vector lengths that sum up to N (c. 1, ll. 11-24, 53-60).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify Joy-Krishna to include executing in parallel instructions having SIMD vector lengths that sum up to N because Nguyen teaches fast speed for repetitive tasks (c. 1, ll. 10-25).

26. Claims 12 and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Joy (US006341347B1) and Krishna (US006161173A) in view of Narayanaswami (US005973705A).

Joy and Krishna are relied upon for teachings as discussed above relative to Claim 9.

However, Joy-Krishna do not teach address space of data memory has frame buffer unit and texture memory unit. But, Narayanaswami teaches SIMD graphics processing system having frame buffer unit (frame buffer 110f, Fig. 2A) while implicitly suggesting texture memory unit.

Art Unit: 2628

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify Joy and Krishna so address space of data memory has frame buffer unit and texture memory unit because Narayanaswami teaches it reduces processing time (c. 2, ll. 20-22).

27. Claims 14-16 and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Joffe (US006330584B1) in view of Krishna (US006161173A)

28. As per Claim 14, Joffe teaches executing instructions from plurality of programs comprising identifying N programs of plurality of programs wherein each of the plurality of programs has a plurality of instructions (c. 2, ll. 11-14, 66-67; c. 1, ll. 62-c. 2, ll. 7); interleaving instructions from N programs in processor pipeline (160, Fig. 1; c. 2, ll. 29-34; c. 3, ll. 40-42); and executing instructions such that a first instruction from one of the N programs is completed before beginning execution of a second instruction of the one of the N programs (c. 2, ll. 35-39)

But, Joffe doesn't teach pipeline has average latency of 1 instruction per cycle, checking no no-op is inserted into pipeline for purpose of ensuring 1st instruction is completed before beginning execution of 2nd instruction. But, Krishna teaches this, as discussed for Claim 23.

29. As per Claim 15, Joffe teaches assigning program counter to each program (c. 2, ll. 8-13).

30. As per Claim 16, Joffe teaches assigning register to each of N programs (c. 2, ll. 8-13).

31. As per Claim 33, Joffe teaches executing instructions from plurality of programs (c. 2, ll. 66-67), assigning 1st output register slot to first of plurality of programs wherein each of the plurality of programs has plurality of instructions (c. 1, ll. 62-c. 2, ll. 11). If wait signal is asserted, instruction execution is not completed, so instruction will be executed again until wait signals are deasserted, then next instruction can be executed (c. 10, ll. 20-24, 31-32), and process repeats until all instructions have been executed. Joffe teaches if task attempts to access

Art Unit: 2628

unavailable resource, task is suspended. When resource becomes available, suspended task is resumed, and instruction accessing resource is re-executed. Task does not get access to same resource until after every other task sharing resource has finished accessing resource (c. 2, ll. 29-39). If Wait signal is asserted, instruction execution is not completed and PC register is frozen, but task remains active, and instruction will be executed again starting next clock cycle. If Suspend/Wait signals are deasserted, the PC register is changed to point to next instruction (c. 10, ll. 19-32). So, Joffe teaches checking to see if Suspend/Wait signals are desasserted, which indicates instruction execution is completed (c. 10, ll. 19-32) and resource has become available (c. 2, ll. 32-34), and resource becomes available after every other task sharing resource has finished accessing resource (c. 2, ll. 29-39). So, Joffe teaches checking to see if all of the programs are completed. So, Joffe teaches executing instructions of first program until program is completed; loading output of first program into its reserved space when first program is completed (c. 9, ll. 26-41); checking to see if all of plurality of programs are completed (c. 2, ll. 35-39). Wait signal is asserted if register is not available, and wait signal is deasserted if register is available for new instruction (c. 10, ll. 20-24, 31-32). Each task (program) has separate register and separate flags (c. 2, ll. 11-13). So, 2nd output register slot is assigned to second program. If task attempts to access unavailable resource, task is suspended. When resource becomes available, suspended task is resumed, and instruction accessing resource is executed (c. 2, ll. 29-34). So, Joffe teaches checking to see if 2nd register slot is available to assign to 2nd program from plurality of programs when 1st program is completed; checking to see if one or more instructions are available when at least one of the programs is not completed (c. 2, ll. 35-39).

However, Joffe does not teach placing no-op when no more instructions are available. However, Krishna teaches information in each entry describes either no-op or associated operation which is to be executed (c. 5, ll. 36-38). So, when there is no-op, that means that no more instructions are available. This would be obvious for reasons for Claim 14.

32. Claim 17 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joffe (US006330584B1) and Krishna (US006161173A) in view of Narayanaswami (US005973705A).

Joffe and Krishna are relied upon for the teachings as discussed above relative to Claim 14. Joffe teaches execution pipeline has depth of N (c. 1, ll. 62-65).

However, Joffe and Krishna do not teach pipeline is graphics processing pipeline. However, Narayanaswami teaches graphics processing execution pipeline (c. 1, ll. 27-43).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify devices of Joffe and Krishna so pipeline is graphics pipeline because Narayanaswami suggests graphics processing is usually implemented in pipeline since different operations are usually performed on graphics data in serial manner (110c, 110d, Fig. 2A; c. 1, ll. 27-43; c. 5, ll. 52-56), and so using pipeline for processing of graphics data is well-known in art.

33. Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joffe (US006330584B1) and Krishna (US006161173A) in view of Nguyen (US005961628A).

Claim 18 is similar in scope to Claim 8, and so is rejected under the same rationale.

Allowable Subject Matter

34. Claims 25-31 are allowed, for reasons given in the Office Action dated March 28, 2007.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joni Hsu whose telephone number is 571-272-7785. The examiner can normally be reached on M-F 8am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kee Tung can be reached on 571-272-7794. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

JH



KEE M. TUNG
SUPERVISORY PATENT EXAMINER